An Interop Publication

CONNEXIONS

The Interoperability Report

December 1995 Volume 9, No. 12

ConneXions —
The Interoperability Report tracks current and emerging standards and technologies within the computer and communications industry.

In this issue:

ConneXions is published monthly by Interop Company, a division of SOFTBANK Exposition and Conference Company, 303 Vintage Park Drive, Foster City, California, 94404–1138, USA.
Phone: +1 (415) 578-6900
Fax: +1 (415) 525-0194
E-mail: connexions@interop.com

Subscription hotline: 1-800-575-5717
or +1 610-892-1959

ISSN 0894–5926

From the Editor

As the year draws to a close we take another look at one region of the ever-growing Internet. It has been some time since we last reported on developments in Europe. This month Jon Crowcroft, with the help of a number of contributors, has put together a glossary of the European Internet. More information can be obtained from the many URLs listed at the end of the article.

Speaking of glossaries, I am in the process of updating and revising the "NetWorld+Interop Pocket Glossary of Networking Terms." If you have suggestions for words to include, please send me e-mail. The intention is not to list every computer term or acronym, but rather to focus on those phrases that relate to networking in general and the Internet in particular.

If your responsibilities include maintaining the security of your organization's network you are well aware of the magnitude of such a task. Well publicized break-ins and other kinds of security compromises have focused much attention on this topic. The "lack of security" is often cited by organizations as a reason for not connecting their corporate networks to the global Internet. This is of course an over-simplification, the Internet has a number of security mechanisms that can be employed (firewalls, cryptographic systems, and so on). But technology is only half the answer. You also need to set proper security policies and procedures for your users, and gain their cooperation in order to succeed. Specific Internet security *strategies* and *policies* are discussed in two articles adapted from the book *Building Internet Firewalls* by D. Brent Chapman and Elizabeth D. Zwicky. You will find the first of these articles in this month's issue; the second will appear in our January 1996 issue.

Security is by no means a "solved problem" in the Internet, but efforts are underway in the Internet Engineering Task Force (IETF) to define a number of security enhancements to the existing architecture. Ted Doty describes some of these technologies in an article entitled "Towards Real Internet Security."

And with that we come to the end of Volume 9 of *ConneXions*. We would like to extend our warmest holiday wishes to all our readers and hope you will continue to send us your comments and suggestions in the new year. As always, you can reach us most easily by e-mail to **connexions@interop.com.**

A Brief Glossary/Overview of the European Internet

by Jon Crowcroft, University College London
(*with help from many others*)

Introduction

This is an overview of the European Internet scene as sampled in the autumn of 1995. Disclaimer: It is necessarily incomplete and inaccurate as such documents always are. Contributions and corrections should be mailed to Jon Crowcroft (`J.Crowcroft@cs.ucl.ac.uk`). A version of this text is available at URL:

`http://www.cs.ucl.ac.uk/staff/jon/euro.html`

The European Internet scene

The scene for Internet access in Europe is quite complex. This is primarily due to the incomplete deregulation of the (until recently) largely state-run monopoly (in the sense of exclusive single provider) telecommunications companies. Part of the goal of the EC is to provide a more streamlined economy, and the high prices and restrictive practices and restricted services that PNOs (*Public Network Operators*—formerly known as PTT [Post, Telephone and Telegraph]) exemplify seem a primary target (along with airlines!) Unfortunately, the deadline for full deregulation is still 2 years away, and the process is relatively slow. To put one data point on the graph, it is public knowledge that BT (the UK PTT, the first large EC nation to deregulate) is paying £ 700M per annum in redundancy pay 10 years after it started. This is only just sustainable, and only really for France and Germany, where their business is relatively large. In only 10 years, the UK has seen nearly an order of magnitude cheaper data communication than the rest of Europe on average, while we have nearly 20 licensed telecom outfits (as well as untold VAN providers, such as Internet Services).

[The history of costs and services in Scandinavia has been very different, and the fact that Finland has the highest percentage per capita with access to the Internet in the world is a clear indicator of this].

In the face of high leased line charges, the growth in domestic Internet access outside of the UK has been close to zero.

Research

In fact, the research community in Europe has had a more mixed success. The shear buying power of the large institutes such as CERN and the (largely state run) University teaching and research community has meant that we have been able to build pan-Euroean Internet access from modest beginnings to something not far different in performance than the NSFNET shortly before its recent demise. However, the provision has been very patchy, and never quite as good as the underlying capacity should permit. For example, a long predilection for arcane protocols such as X.25 led to a period when a 2Mbps backbone was run over a switched network with less than wonderful success until various window and packet size parameters in the underlying X.25 were set to make the lower level as transparent as possible.

Meanwhile, plans are afoot to build the successor to this. Currently, the EC has perceived the requirement to coordinate electronic communications for European research as a massive cost-savings exercise (given the travel budget of any RACE or Esprit project it surprises me that they hadn't spotted this one before—or else concluded a similar thing about airlines). Thus in the pipeline are a number of proposals that will bid to provide 34Mbps, 155Mbps and eventually 622Mbps (or more) between national research and educational networks.

Some of the proposals are based (predictably) on ATM technology. Others are more thoughtfully reevaluating the experience from some national trials (as well as the US Gigabit projects) and wondering whether a more hybrid approach might not be more sensible (or even just a set of high speed point-to-point links between multi-protocol routers).

At the time of writing, it is not clear which project(s) will get definite funding, nor what their lifetime will be. All such efforts have to eventually lead to something being provided on a more commercial footing (as happened in the US).

Spare lines

Another datapoint that is worth noting is that there are around 45Gbps worth of fiber trunking spare in the 3 largest EC nations—with current tariffs, the PNOs simply cannot sell this capacity. With a research project they can experiment with new services without being subject to the dangers of undermining their own pure commercial business, and it can give them an edge into providing more up-to-date services (perhaps even move into Information Services, since transmission capacity, in the long run, is going to be a very marginal business).

While the decisions are being made, other companies are leasing lines from the telcos, slinging up meshes of routers, and using dial-up (often ISDN) for "subscriber loop" Internet access and making a profit! This must be galling to a shareholder in a telco who put in all the cabling infrastructure to make this possible—on the other hand, due to inflated local call charges, they often do not care since they make more money from the long call times people run up browsing the ever-slower Web!

We live in interesting times.

Glossary

The items in this network glossaryy fall into 3 basic categories. These are given in parenthesis following each entry. Key: 1= Network, 2=Organisation (ISP or other), 3=Unknown.

ACOnet (1) Austrian Academic Computer Network. The national R&D network which provides services to all universities, some research centers, as well as to educational, cultural and governmental sites in Austria.

Ariadnet (1) The Greek National Internet service.

ARNES (1) The Academic and Research Network of Slovenia

BELNET (1) The Belgian Research Network.

CARnet (1) A Croatian Internet service.

CESnet (1) The Academic and Research network in the Czech Republic.

BTnet (1) *British Telecom Network*. Not to be confused with IBDNS, which is also operated by BT.

CERN (2) European Laboratory for Particle Physics. Birthplace of the World-Wide Web.

Clinet (2) A small ISP ("microprovider") located in Helsinki.

Cylink (2) EUNet in Cyprus.

continued on next page

Glossary of the European Internet *(continued)*

DANTE (2) *Delivering Advanced Network Technology to Europe.* See EuropaNet.

Datanet (1) Commercial LAN interconnect service offered by Telecom Finland.

DAXnet (2) Norwegian consulting company also offering Internet connectivity.

DFN (1) *Deutsches Forschungsnetz.* The German national research and academic network.

DENet (1) The Danish research network, operated by UNI-C.

EARN (3) *European Academic and Research Network.* Merged with RARE in 1994 to form TERENA.

EBONE (1) The *European Backbone*—one of the leading pan-European service providers.

ECRC (2) German industrial research centre in München.

EENet (1) The Estonian Educational and Research Network.

EMPB (1) *European Multi Protocol Backbone.* Previous EuropaNET (DANTE) backbone, operated by Unisource.

EU (3) *The European Union.* Major obstacle for free competitive networking in Europe.

EUnet (2) Currently, (10/95) the largest commercial ISP in Europe.

EuropaNET (1) DANTE's European network service. Currently using BT's IBDNS as its European backbone. On 1 October 1995 BT launched IBDNS, the successor of EMPB, the Western European backbone component of EuropaNET, DANTE's international network service for the European R&D community. EuropaNET was first launched in October 1992. DANTE awarded the contract for the replacement of the backbone to BT in May 1995 following a public invitation to tender. Under the contract, BT provides access services at speeds up to 8 Mbps to networks in 15 European countries.

FICIX (3) *Finnish Commercial Internet Exchange.* An exchange point for EUnet Finland, Datanet (Telecom Finland), LanLink (FINNET group) and FUNET.

FUNET (1) The *Finnish University Network.* The academic computer network in Finland.

FORTHnet (1) *FOundation for Research and Technology Hellas.*

GARR (1) The Italian Research Network. (*Gruppo Armonizzazione Reti della Ricerca*).

GIX (3) *Global Internet eXchange.* Note: some confusion may arise here, if you see the term "the GIX," this usually means the MAE-East Internet exchange point in Washington DC, USA, which could also be referred to as a "de-facto NAP." There has been plans to physically distribute the GIX and thereby create a "D-GIX." The technical/strategic plan is to create a market for providing bandwidth at layer 2 between the individual GIX installations. However, this has not yet happened, at least not as an operational infrastructure. The internet exchange located at KTH is however usually referred to as "the Stockholm D-GIX" or "the D-GIX at KTH" or possibly simply "the D-GIX" (which could be confusing).

HEAnet (1) The *Higher Education Authority's Network* in Ireland.

HUNGARNET (1 and 2) The association and computer network of Hungarian institutes of higher education, research and development, libraries and other public collections.

IBR (1) The "IBR-LAN" is where the Amsterdam Internet Exchange is currently living; don't know where the name comes from.

INRIA (2) French national research institution/organisation; where EUnet France saw the light of day many years ago. *Institut National de Recherche en Informatique et en Automatique.* Organisation currently in charge of FR NIC (not an ISP). INRIA is an EPST (*Etablissement Public a caractere Scientifique et Technique*) as CNRS.

InterBusiness (2) An Italian commercial Internet Provider.

Internet Way (2) A French Commercial Internet Provider.

IntIS (2) Internet Iceland Inc. Limited company that runs the Icelandic Internet, ISnet. Formally part of NORDUnet. This is the successor of SURIS, which used to be the academic and research network in Iceland.

ISO (2) The *International Organization for Standardization* (ISO is a multilingual short form name, not an acronym).

ITU (2) Formally the CCITT, the International Telecommunications Union—Has several sectors, standardisation and regulation.

KTH (2) The Royal Institute of Technology in Stockholm (really *Kungliga Tekniska Högskolan*).

JANET (1) The *Joint Academic Network*—the UK Educational and Research network run by UKERNA.

LanLink (1) Commercial LAN interconnect (IP, DECnet, IPX, etc.) service offered by FINNET group (a consortium of local PNOs).

LATNET (1) The Latvian Academic and Research Network.

LINX (3) *London INternet Exchange*—an Internet interconnect point at Telehouse, London, where all UK and many European providers exchange traffic.

LITNET (1) The Lithuanian University and Research Network.

MFS (2) *Metropolitan Fiber Systems.* They are perhaps most known in our circles for operating the physical infrastructure of the MAE- East Internet exchange, and are also involved in MAE-West. This is a PNO-style organization with its outspring in the US, however they are beginning to establish themselves with a presence in cities in Europe as well.

MSU (1) Moscow State University's network.

NASK (1) Research and Academic Network in Poland (*Naukowa i Akademicka Siec Komputerowa*).

NIKHEF (2) *Nationaal Instituut voor Kern- en Hoge-Energie Fysica.* Dutch institute for nuclear and high-energy physics.

NORDUnet (2) International network operator, provides services the Nordic national networks and to other networks of interest to the company and its owners: DENet (Denmark), FUNET (Finland), SURIS (Iceland), UNINETT (Norway), SUNET (Sweden).

continued on next page

Glossary of the European Internet *(continued)*

Official wording: NORDUnet is an international network operator that provides services and international connectivity to the Nordic National Networks in Denmark, Finland, Iceland, Norway and Sweden. In many aspects NORDUnet operates as the international branch of these networks.

Oslonett (2) Commercial Internet service provider in Norway, with "home base" in Oslo (as the name should indicate). Currently has a large chunk of the dial-up Internet access market in Norway.

PING (1) A Swiss commercial Internet provider.

PIPEX (2) *Public IP EXchange.* The largest commercial Internet service provider in the UK, and one of a number of Pan-European providers. A subsidiary of Unipalm Group plc.

Planete.net (2) A French commercial Internet provider.

The Planet Online, Ltd (2) A digital dial-on-demand (using ISDN) service provider based in Leeds, UK.

PowerTech (2) Commercial Internet service provider in Norway, also in Oslo.

PTTs (2) Post, Telephone, Telegraph. Telecom companies—now known as *PNOs.*

PNO (2) *Public Network Operator.* Usually a PTT of some sort.

PROF-I-NET (2) An Austrian Information Provider.

RAIN (2) France Telecom Transpac's name for their commercial Internet service.

RARE (3) *Réseaux Associés pour la Recherche Européenne.* (The Association of European Research Networks and their users). Merged with EARN in 1994 to form TERENA.

RCCN (1) The portuguese Scientific national network. (*Rede da Comunidade Cientifica Nacional*).

RedIRIS (1) The Spanish network for research and development.

Renater (1) *Réseau National de telecommunications pour la Technologie, l'Enseignement et la Recherche.* The French national network for education and research.

RESTENA (1) *Riseau Tiliinformatique de l'Education Nationale et de la Recherche.* Luxembourg's educational and research network.

RIPE (2) a collaborative organisation which consists of European Internet service providers. It is an open and voluntary organisation. It aims to provide the necessary administrative and technical coordination to allow the operation of a seamless pan-European IP network. RIPE does not operate a network of its own. The IP activity of TERENA is RIPE. The RIPE NCC provides a document that is a registry of Internet Providers in Europe.

RIPE NCC (2) The *RIPE Network Coordination Centre* supports all those RIPE activities which cannot be effectively performed by volunteers from the participating organisations. Besides supporting RIPE activities in general the NCC provides a number of services to Internet service providers and network operators across Europe.

SANET The Slovakian Academic network.

SUNET (1) The *Swedish University Network.*

SuperJANET (1) The latest phase in the development of JANET, the UK educational and research network run by UKERNA. It uses SMDS and ATM to provide multi-service network facilities for many new applications including Multimedia Conferencing.

SURFnet (2) Dutch ISP for research and education and other non-profit communities.

SWIPnet (2) Commercial internet service provider in Sweden. Daughter company of Tele2, one of the PNOs in Sweden.

SWITCH (2) The Swiss Academic and Research network.

Telia (2) One of the PNOs in Sweden. This PNO was at one time the only PNO in Sweden, and is (AFAIK) by far the largest in the current Swedish "general" PNO market.

Ten34 (3) Trans-European Network Interconnect at 34–155 Mbps. EU funded project, coordinated by DANTE. Originally ten countries focussing on 34Mbps ATM, now broader perspective and membership.

TERENA (2) *Trans-European Research and Education Networking Association.* Formed in 1994 by a merger of RARE and EARN. For a list of all of the European research networking organisations, you might want to look at the TERENA membership.

TINET (2) A Swiss Internet Provider.

Transpac (1) The IP service operated by France Telecom Network Services AB. For more information.

TUVAKA (2) Turkish networks and organizations TR-NET (network and organization)

UKERNA (2) *UK Education and Research Networking Association.*

Uninett (2) The Norwegian University Network. See also NORDUnet.

Unisource (3) Loosely organised PNO alliance between CH, ES, NL, SE PNOs.

VUCC (2) *Vienna University Computer Center.* (Vienna University is the biggest Austrian university). In addition to caring for the local networking setup, VUCC is tasked with operations and maintenance of ACOnet, as well as with the operations of the EBONE node in Vienna (aka Vienna-EBS) and a sizable set of international links to central and eastern Europe (Poland, Czechia, Slovakia, Hungary, Romania, Bulgaria, Croatia, Slovenia, Macedonia (FYROM)).

Xlink (2) *eXtended Local Informatics Network Karlsruhe.* A German ISP.

Contributors (in no particular order)

Håvard Eidnes	`Havard.Eidnes@runit.sintef.no`
Attila Ozgit	`ozgit@metu.edu.tr`
Steven Bakker	`Steven.Bakker@dante.org.uk`
Petri Ojala	`ojala@eunet.fi`
Marius Olafsson	`marius@rhi.hi.is`
Dave Morton	`Dave.Morton@ecrc.de`
Frode Greisen	`Frode.Greisen@uni-c.dk`
Josefien Bersee	`J.Bersee@dante.org.uk`
Wilfried Woeber	`woeber@cc.univie.ac.at`
John Martin	`martin@terena.nl`
Peter Villemoes	`Peter.Villemoes@nordu.net`

continued on next page

Glossary of the European Internet *(continued)*

Christian Panigl	panigl@cc.univie.ac.at
Keith Mitchell	keith@pipex.com
Lars-Johan Liman	liman@sunet.se
Paolo Bevilacqua	pab@uni.net
Eric Malmström	eric@transpac.net
Per Gregers Bilse	bilse@EU.net
Annie Renard	nic@nic.fr
Prof. Dr. W. Zorn	zorn@ira.uka.de
Francis Dupont	Francis.Dupont@inria.fr

URLs of interest

http://www.aco.net/
http://ithaki.servicenet.ariadne-t.gr/default.html
http://www.arnes.si
http://www.belnet.be
http://www.carnet.hr/
http://www.cesnet.cz
http://www.bt.net/
http://www.CERN.ch/
http://www.Cyprus.EU.net/
http://www.dante.net/welcome/nn-servers.htm
http://www.dante.net/links.html
http://www.dfn.de
http://info.denet.dk/
http://www.uni-c.dk/
http://www.terena.nl/
http://www.ebone.net/
http://www.ecrc.de/networking/
http://www.eenet.ee/
http://www.echo.lu/
http://www.EU.net/
http://www.funet.fi/
http://www.forthnet.gr/
http://www.nis.garr.it/
http://web.hea.ie
http://www.iif.hu/hungarnet.html
http://www.inria.fr
http://www.interbusiness.it/
http://www.iway.fr/
http://www.isnet.is
http://www.itu.ch/
http://www.kth.se/
http://www.ukerna.ac.uk/
http://www.latnet.lv/LATNET/
http://www.linx.net/linx/
http://neris.mii.lt/litnet.html
http://www.mfs.net/
http://www.phys.msu.su/runnet/runnet.html
http://www.nask.org.pl/
http://www.nikhef.nl/
http://www.nordu.net/
http://www.denet.dk/
http://www.funet.fi/
http://www.isnet.is/
http://aun.uninett.no/
http://www.sunet.se/
http://www.ping.ch/
http://www.unipalm.pipex.com
http://www.pipex.net/network/connectivity.html
http://www.planete.net/
http://www.theplanet.net
http://www.theplanet.net/netplan.htm
http://www.plus.at/

```
http://web.transpac.fr/
http://www.terena.nl/
http://www.fccn.pt/RCCN/
http://www.rediris.es/
http://www.urec.fr/Renater/
http://www.restena.lu/
http://www.ripe.net/
http://www.ripe.net/ripe/ncc.html
http://www.sanet.sk/sanet.html
http://www.sunet.se/
http://www.ukerna.ac.uk
http://www.ucs.ed.ac.uk/~jaw/vconf.html
http://www.surfnet.nl/
http://www.swip.net/
http://www.switch.ch/
http://www.telia.se/
http://www.dante.net/ten34.html
http://www.terena.nl/
http://www.tinet.ch/
http://www.transpac.se/
http://www.tr-net.net.tr
http://www.ukerna.ac.uk
http://www.uninett.no/
http://www.nordu.net/
http://www.unisource.se/
http://www.univie.ac.at/
http://www.xlink.net/
```

References

[1] Crowcroft, J., "International Internetworking," *ConneXions,* Volume 2, No. 4, April 1988.

[2] Goldstein, S. & Michau, C., "Convergence of European and North American Research and Academic Networking," *ConneXions,* Volume 5, No. 4, April 1991.

[3] Stockman, B., "Current Status on Networking in Europe," *ConneXions,* Volume 5, No. 7, July 1991.

[4] Stockman, B., "EBONE, The European Internet Backbone," *ConneXions,* Volume 7, No. 5, May 1993.

[5] Stockman, B., "Global Connectivity: The Global Internet Exchange (GIX)," *ConneXions,* Volume 7, No. 11, November 1993.

[6] Réseaux Associés pour la Recherche Européenne, *ConneXions,* Volume 6, No. 1, January 1992.

[7] Bersee, J., "Profile: DANTE and EuropaNET," *ConneXions,* Volume 8, No. 6, June 1994.

[8] Karrenberg, Daniel, "The RIPE NCC and the Routing Registry for Europe," *ConneXions,* Volume 7, No. 11, November 1993.

[9] *ConneXions,* Volume 7, No. 5, May 1993, "Special Issue: Focus on Europe."

[10] Kowack, G., "Profile: EUnet," *ConneXions,* Volume 7, No. 11, November 1993.

JON CROWCROFT is a senior lecturer in the Department of Computer Science, University College London, where he is responsible for a number of European and US funded research projects in Multi-media Communications. He has been working in these areas for over 14 years. He graduated in Physics from Trinity College, Cambridge University in 1979, and gained his MSc in Computing in 1981, and PhD in 1993. He is a member of the ACM, the British Computer Society and the IEE. He was general chair for the ACM SIGCOMM 94 symposium. He is also on the editorial teams for the *Transactions on Networks* and the *Journal of Internetworking.* E-mail: J.Crowcroft@cs.ucl.ac.uk

Internet Security Strategies

by
Brent Chapman, Great Circle Associates
and
Elizabeth Zwicky, Silicon Graphics

This article discusses some of the basic strategies employed in building firewalls and in enforcing security at your site. These are not staggering revelations; they are straightforward approaches. They're presented here so that you can keep them in mind as you put together a firewall solution for your site.

Least Privilege

Perhaps the most fundamental principle of security (any kind of security, not just computer and network security) is that of *least privilege*. Basically, the principle of least privilege means that any object (user, administrator, program, system, whatever) should have only the privileges the object needs to perform its assigned tasks—and no more. Least privilege is an important principle for limiting your exposure to attacks and for limiting the damage caused by particular attacks.

Some car manufacturers set up their locks so that one key works the doors and the ignition, and a different key works the glove compartment and the trunk; that way, you can enforce least privilege by giving a parking lot attendant the ability to park the car without the ability to get at things stored in the trunk. Many people use splittable key chains, for the same reason. You can enforce least privilege by giving someone the key to your car, but not the key to your house as well.

In the Internet context, the examples are endless. Every user probably doesn't need to access every Internet service. Every user probably doesn't need to modify (or even read) every file on your system. Every user probably doesn't need to know the machine's root password. Every system administrator probably doesn't need to know the root passwords for all systems. Every system probably doesn't need to access every other system's files.

Applying the principle of least privilege suggests that you should explore ways to reduce the privileges required for various operations. For example:

- Don't give a user the root password for a system if all she needs to do is reset the print system. Instead, write a privileged program the user can run that resets the print system.
- Don't make a program run `setuid` to root if all it needs to do is write to one protected file. Instead, make the file group-writable to some group and make the program run `setgid` to that group rather than `setuid` to root.
- Don't have your internal systems trust one of your firewall machines just so it can do backups. Instead, make the firewall machine trust the internal system, or, better yet, put a local tape drive on the firewall machine so that it can do its own backups.

Many of the common security problems on the Internet can be viewed as failures to follow the principle of least privilege. For example, there have been and continue to be any number of security problems discovered in *Sendmail,* which is a big, complex program; any such program is going to have bugs in it. The problem is that *Sendmail* runs (at least some of the time) `setuid` to root; many of the attacks against *Sendmail* take advantage of this.

Because it runs as root, *Sendmail* is a high-value target that gets a lot of attention from attackers; the fact that it's a complex program just makes their jobs easier. This implies both that privileged programs should be as simple as possible and that, if a complex program requires privileges, you should look for ways to separate and isolate the pieces that need privileges from the complex parts. (It is important to realize that *Sendmail* is far from the only example we could cite; you can find similar problems in almost any large, complex, privileged piece of software.)

Many of the solutions you'll employ in protecting your site are tactics for enforcing the strategy of least privilege. For example, a packet filtering system is designed to allow in packets for the services you want. Running insecure programs in an environment where only the privileges the programs absolutely need are available to them (e.g., a machine that's been stripped down in one way or another) is another example; this is the essence of a *bastion host*.

There are two problems with trying to enforce least privilege. First, it can be complex to implement when it isn't already a design feature of the programs and protocols you're using. Trying to add it on may be very difficult to get right. Some of the cars that try to implement least privilege with separate keys for the trunk and the ignition have remote trunk release buttons that are accessible without the keys, or fold-down rear seats that allow you to access the trunk without opening it the traditional way at all. You need to be very careful to be sure that you've actually succeeded in implementing least privilege.

Second, you may end up implementing something less than least privilege. Some cars have the gas cap release in the glove compartment. That's intended to keep parking lot attendants from siphoning off your gas, but if you lend a friend your car, you probably want them to be able to fill it up with gas. If you give your friend only the ignition key, you're giving them less than the minimum privilege you want them to have (because they won't be able to fill up the gas tank), but adding the key to the trunk and the glove compartment may give them more privilege than you want them to have.

You may find similar effects with computer implementations of least privilege. Trying to enforce least privilege on people, rather than programs, can be particularly dangerous. You can predict fairly well what permissions *Sendmail* is going to need to do its job; human beings are less predictable, and more likely to become annoyed and dangerous if they can't do what they want to. Be very careful to avoid turning your users into your enemies.

Defense in depth

Another principle of security (again, any kind of security) is *defense in depth*. Don't depend on just one security mechanism, however strong it may seem to be; instead, install multiple mechanisms that back each other up. You don't want the failure of any single security mechanism to totally compromise your security. You can see applications of this principle in other aspects of your life. For example, your front door probably has both a doorknob lock and a deadbolt; your car probably has both a door lock and an ignition lock; and so on.

Although the focus of our book (from which this article is adapted) is on firewalls, we don't pretend that firewalls are a complete solution to the whole range of Internet security problems. Any security—even the most seemingly impenetrable firewall—can be breached by attackers who are willing to take enough risk and bring enough power to bear.

continued on next page

Internet Security Strategies *(continued)*

The trick is to make the attempt too risky or too expensive for the attackers you expect to face. You can do this by adopting multiple mechanisms that provide backup and redundancy for each other: network security (a firewall), host security (particularly for your bastion host), and human security (user education, careful system administration, etc.). All of these mechanisms are important and can be highly effective, but don't place absolute faith in any one of them.

Your firewall itself will probably have multiple layers. For example, one architecture has multiple packet filters; it's set up that way because the two filters need to do different things, but it's quite common to set up the second one to reject packets that the first one is supposed to have rejected already. If the first filter is working properly, those packets will never reach the second; however, if there's some problem with the first, then hopefully you'll still be protected by the second. Here's another example: if you don't want people sending mail to a machine, don't just filter out the packets, also remove the mail programs from the machine. In situations where the cost is low, you should always employ redundant defenses.

Choke Point

A *choke point* forces attackers to use a narrow channel, which you can monitor and control. There are probably many examples of choke points in your life: the toll booth on a bridge, the check-out line at the supermarket, the ticket booth at a movie theater.

In network security, the firewall between your site and the Internet (assuming that it's the only connection between your site and the Internet) is such a choke point; anyone who's going to attack your site from the Internet is going to have to come through that channel, which should be defended against such attacks. You should be watching carefully for such attacks and be prepared to respond if you see them.

A choke point is useless if there's an effective way for an attacker to go around it. Why bother attacking the fortified front door if the kitchen door around back is wide open? Similarly, from a network security point of view, why bother attacking the firewall if there are dozens or hundreds of unsecured dial-up lines that could be attacked more easily and probably more successfully?

A second Internet connection—even an indirect one, like a connection to another company which has its own Internet connection elsewhere—is an even more threatening breach. Internet-based attackers might not have a modem available, or might not have gotten around to acquiring phone service they don't need to pay for, but they can certainly find even roundabout Internet connections to your site.

A choke point may seem to be putting all your eggs in one basket, and therefore a bad idea, but the key is that it's a basket you can guard carefully. The alternative is to split your attention among many different possible avenues of attack. If you split your attention in this way, chances are that you won't be able to do an adequate job of defending any of the avenues of attack, or that someone will slip through one while you're busy defending another (where they may even have staged a diversion specifically to draw your attention away from their real attack).

Weakest Link

A fundamental tenet of security is that a chain is only as strong as its *weakest link* and a wall is only as strong as its weakest point. Smart attackers are going to seek out that weak point and concentrate their attentions there.

You need to be aware of the weak points of your defense so that you can take steps to eliminate them, and so that you can carefully monitor those you can't eliminate. You should try to pay attention evenly to all aspects of your security, so that there is no large difference in how insecure one thing is as compared to another.

There is always going to be a weakest link, however; the trick is to make that link strong enough and to keep the strength proportional to the risk. For instance, it's usually reasonable to worry more about people attacking you over the network than about people actually coming to your site to attack you physically; therefore you can usually allow your physical security to be your weakest link. It's not reasonable to neglect physical security altogether, however, because there's still some threat there. It's also not reasonable, for example, to protect *Telnet* connections very carefully, but not protect FTP connections, because of the similarities of the risks posed by those services.

Host security models suffer from a particularly nasty interaction between choke points and weak links; there's no choke point, which means that there are a very large number of links, and many of them may be very weak indeed.

Fail-Safe Stance

Another fundamental principle of security is that, to the extent possible, systems should *fail safe;* that is, if they're going to fail, they should fail in such a way that they deny access to an attacker, rather than letting the attacker in. The failure may also result in denying access to legitimate users as well, until repairs are made, but this is usually an acceptable tradeoff.

Safe failures are another principle with wide application in familiar places. Electrical devices are designed to go off—to stop—when they fail in almost any way. Elevators are designed to grip their cables if they're not being powered. Electric door locks generally unlock when the power fails, to avoid trapping people in buildings.

Most of the applications we discuss automatically fail safely. For example, if a packet filtering router goes down, it doesn't let any packets in. If a proxying program goes down, it provides no service. On the other hand, some host-based packet filtering systems are designed such that packets are allowed to arrive at a machine that runs a packet filtering application and separately runs applications providing services. The way some of these systems work, if the packet filtering application crashes (or is never started at boot time), the packets will be delivered to the applications providing services. This is not a fail-safe design and should be avoided.

The biggest application of this principle in network security is in choosing your site's *stance* with respect to security. Your stance is, essentially, your site's overall attitude towards security. Do you lean towards being restrictive or permissive? Are you more inclined to err in the direction of safety (some might call it paranoia) or freedom?

There are two fundamental stances that you can take with respect to security decisions and policies:

- The default deny stance: Specify only what you allow and prohibit everything else.
- The default permit stance: Specify only what you prohibit and allow everything else.

continued on next page

Internet Security Strategies *(continued)*

It may seem obvious to you which of these is the "right" approach to take; from a security point of view, it's the default deny stance. Probably, it will also seem obvious to your users and management; from their point of view, it's the default permit stance. It's important to make your stance clear to users and management, as well as to explain the reasons behind that stance. Otherwise, you're likely to spend a lot of unproductive time in conflict with them, wondering "How could they be so foolish as to even suggest that?" time and again, simply because they don't understand the security point of view.

Default Deny Stance:

"That Which Is Not Expressly Permitted Is Prohibited." The *default deny stance* makes sense from a security point of view because it is a fail-safe stance. It recognizes that what you don't know *can* hurt you. It's the obvious choice for most security people, but it's usually not at all obvious to users.

With the default deny stance, you prohibit everything by default; then, to determine what you are going to allow, you:

- Examine the services your users want.
- Consider the security implications of these services and how you can safely provide them.
- Allow only the services that you understand, can provide safely, and see a legitimate need for.

Services are enabled on a case-by-case basis. You start by analyzing the security of a specific service, and balance its security implications against the needs of your users. Based on that analysis and the availability of various remedies to improve the security of the service, you settle on an appropriate compromise.

For one service, you might determine that you should provide the service to all users and can do so safely with commonly available packet filtering or proxy systems. For another service, you might determine that the service cannot be adequately secured by any currently available means, but that only a small number of your users or systems require it. In the latter case, perhaps its use can be restricted to that small set of users (who can be made aware of the risks through special training) or systems (which you may be able to protect in other ways; e.g., through host security). The whole key is to find a compromise that is appropriate to your particular situation.

Default Permit Stance:

"That Which Is Not Expressly Prohibited Is Permitted." Most users and managers prefer the *default permit stance.* They tend to assume that everything will be, by default, permitted, and that certain specific, troublesome actions and services will then be prohibited as necessary. For example:

- NFS is not permitted across the firewall.
- WWW access is restricted to users who have received awareness training about its security problems.
- Users are not allowed to set up unauthorized servers.

They want you to tell them what's dangerous; to itemize those few (they think) things that they can't do, and to let them do everything else. This is definitely not a fail-safe stance.

First, it assumes that you know ahead of time precisely what the specific dangers are, how to explain them so users will understand them, and how to guard against them. Trying to guess what dangers might be in a system or out there on the Internet is essentially an impossible task. There are simply too many possible problems, and too much information (e.g., new security holes, new exploitations of old holes, etc.) to be able to keep up to date. If you don't know that something is a problem, it won't be on your "prohibited" list. In that case, it will go right on being a problem until you notice it and you'll probably notice it because somebody takes advantage of it.

Second, the default permit stance tends to degenerate into an escalating "arms race" between the firewall maintainer and the users. The maintainer prepares defenses against user action or inaction (or, he just keeps saying, "Don't do that!"); the users come up with fascinating new and insecure ways of doing things; and the process repeats, again and again. The maintainer is forever playing catch up. Inevitably, there are going to be periods of vulnerability between the time that a system is set up, the time that a security problem is discovered, and the time that the maintainer is able to respond to the problem. No matter how vigilant and cooperative everyone may be, some things are going to fall through the cracks forever: because the maintainer has never heard about them, because he has never realized the full security consequences; or because he just plain hasn't had time to work on the problem.

About the only people who benefit from the default permit stance are potential attackers, because the firewall maintainer can't possibly close all the holes, is forever stuck in "fire fighting" mode, and is likely to be far too busy to notice an attacker's activities.

For example, consider the problem of sharing files with collaborators at another site. Your users' first idea will probably be to use the same tool that they use to share files internally—NFS. The problem is, NFS is completely unsafe to allow across a firewall (for reasons discussed elsewhere in our book). Suppose that your stance is a permissive one, and you haven't specifically told your users that it's not safe to run NFS across your firewall (or even if you have told them, but don't remember or don't care). In this case, you're probably going to find yourself running NFS across your firewall because it seemed like a good idea to somebody who didn't understand (or care about) the security issues. If your stance is default deny, on the other hand, your users' attempts to set up NFS will fail. You'll need to explain why to them, suggest alternatives that are more secure (such as FTP), and look for ways to make those more secure alternatives easier to use without sacrificing security.

Universal participation

In order to be fully effective, most security systems require the *universal participation* (or at least the absence of active opposition) of a site's personnel. If someone can simply opt out of your security mechanisms, then an attacker may be able to attack you by first attacking that exempt person's system and then attacking your site from the inside. For example, the best firewall in the world won't protect you if someone who sees it as an unreasonable burden sets up a back-door connection between your site and the Internet in order to circumvent the firewall. This can be as easy as buying a modem, obtaining free PPP or SLIP software off the Internet, and paying a few dollars a month to a local low-end Internet service provider; this is well within the price range and technical abilities of many users and managers.

continued on next page

Internet Security Strategies *(continued)*

Much more mundane forms of rebellion will still ruin your security. You need everybody to report strange happenings that might be security-related; you can't see everything. You need people to choose good passwords; to change them regularly; and not to give them out to their friends, relatives, and pets.

How do you get everyone to participate? Participation might be voluntary (you convince everybody that it's a good idea) or involuntary (someone with appropriate authority and power tells them to cooperate or else), or some combination of the two. Obviously, voluntary participation is strongly preferable to involuntary participation; you want folks helping you, not looking for ways to get around you. This means that you may have to work as an evangelist within your organization, selling folks on the benefits of security and convincing them that the benefits outweigh the costs.

People who are not voluntary participants will go to amazing lengths to circumvent security measures. On a voicemail system that required passwords to be changed every month, numerous people discovered that it recorded only six old passwords, and took to changing their passwords seven times in a row (in seven separate phone calls!) in order to be able to use the same password. This sort of behavior leads to an arms race (the programmers limit the number of times you can change your password), and soon numerous people are sucked into a purely internal battle. You have better things to do with your time, as do your users; it's worth spending a lot of energy to convince people to cooperate voluntarily, because you'll often spend just as much to force them, with worse side effects.

Diversity of defense

Just as you may get additional security from using a number of different systems to provide depth of defense, you may also get additional security from using a number of different types of systems. If all of your systems are the same, somebody who knows how to break into one of them probably knows how to break into all of them.

The idea behind *diversity of defense* is that using security systems from different vendors may reduce the chances of a common bug or configuration error that compromises them all. There is a tradeoff in terms of complexity and cost, however. Procuring and installing multiple different systems is going to be more difficult, take longer, and be more expensive than procuring and installing a single system (or even several identical systems). You're going to have to buy the multiple systems (at reduced discounts from each vendor, because you're buying less from them) and multiple support contracts to cover them. It's also going to take additional time and effort for your staff to learn how to deal with these different systems.

Beware of illusionary diversity. Simply using different vendors' UNIX systems probably won't buy you diversity, because most UNIX systems are derived from either the BSD or System V source code. Further, most common UNIX networking applications (such as *Sendmail, telnet/telnetd, ftp/ftpd,* and so on), are derived from the BSD sources, regardless of whether they're on a BSD- or System V-based platform. There were any number of bugs and security problems in the original releases that were propagated into most of the various vendor-specific versions of these operating systems; many vendor-specific versions of UNIX still have bugs and security problems that were first discovered years ago in other versions from other vendors, and have not yet been fixed.

Also beware that diverse systems configured by the same person (or group of people) may share common problems if the problems stem from conceptual rather than technological roots. If the problem is a misunderstanding about how a particular protocol works, for example, your diverse systems may all be configured incorrectly in the same way according to that misunderstanding.

Although many sites acknowledge that using multiple types of systems could potentially increase their security, they often conclude that diversity of defense is more trouble than it's worth, and that the potential gains and security improvements aren't worth the costs. We don't dispute this; each site needs to make its own evaluation and decision concerning this issue.

Simplicity

Simplicity is a security strategy for two reasons. First, keeping things simple makes them easier to understand; if you don't understand something, you can't really know whether or not it's secure. Second, complexity provides nooks and crannies for all sorts of things to hide in; it's easier to secure a studio apartment than a mansion.

Complex programs have more bugs, any of which may be security problems. Even if bugs aren't in and of themselves security problems, once people start to expect a given system to behave erratically, they'll accept almost anything from it, which kills any hope of their recognizing and reporting security problems with it when these problems do arise.

References

[1] Garfinkel, Simson & Spafford, Gene, *Practical UNIX and Internet Security,* O'Reilly & Associates, 1996, ISBN 1-56592-148-8.

[2] Holbrook, P. and Reynolds, J., "Site Security Handbook," RFC 1244, July 1991.

[3] Ranum, Marcus, "Internet Firewalls Frequently Asked Questions (FAQ)." Available from: `http://www.iwi.com/pubs/faq.htm`

[4] Ranum, Marcus, "Thinking About Firewalls," 1993. Available from: `ftp://moink.nmsu.edu/firewalls/fwalls.ps.Z`

[5] See also the `comp.admin.policy` newsgroup.

[6] Doty, T., "The Firewall Heresies," *ConneXions,* Volume 9, No. 6, June 1995.

[7] Doty, T., "A Firewall Overview," *ConneXions,* Volume 9, No. 7, July 1995.

[Ed.: This article is adapted from *Building Internet Firewalls* by D. Brent Chapman and Elizabeth D. Zwicky, published by O'Reilly & Associates, 1995, ISBN 1-56592-124-0, 1-800-998-9938. Used with permission.]

Coming in the January 1996 issue of ConneXions:

Internet Security Policies

D. BRENT CHAPMAN is a consultant in the San Francisco Bay Area, specializing in Internet firewalls. He has designed and built many Internet firewall systems for a wide range of clients, using a variety of techniques and technologies. He is the manager of the Firewalls Internet mailing list. Before founding Great Circle Associates, he was operations manager for a financial services company, a world-renowned corporate research lab, a software engineering company, and a hardware engineering company. He holds a Bachelor of Science degree in Electrical Engineering and Computer Science from the University of California, Berkeley. In his spare time, Brent is a volunteer search and rescue pilot, disaster relief pilot, and mission coordinator for the California Wing of the Civil Air Patrol (the civilian auxiliary of the United States Air Force). E-mail: `Brent@GreatCircle.com`

ELIZABETH D. ZWICKY is a senior system administrator at Silicon Graphics, and the president of the System Administrators Guild (SAGE). She has been doing large-scale UNIX system administration for 10 years, and was a founding board member of both SAGE and BayLISA (the San Francisco Bay Area system administrators' group), as well as a non-voting member of the first board of the Australian system administration group, SAGE-AU. She has been involuntarily involved in Internet security since before the Internet Worm. In her lighter moments, she is one of the few people who makes significant use of the "rand" function in *PostScript,* producing *PostScript* documents that are different every time they're printed. E-mail: `zwicky@sgi.com`

Towards Real Internet Security

by Ted Doty, Network Systems Corporation

Introduction

As with most things concerning security and the Internet, there is good news and bad news. The good news is that the Internet is alive an well, still growing like a weed. The bad news is that ruffians are still riding more or less unmolested over the electronic landscape. Break-ins, ranging from joy riding to industrial espionage are a definite growth industry. The problem is *anonymity:* in an environment where technology allows people to mask their true identity, there is little risk associated with anti-social behavior. Some solutions—such as firewalls—have been proposed that attempt to address this. Unfortunately, many of these attempts to eliminate anonymity have serious technical shortcomings, due to limitations in the TCP/IP protocol suite itself [1].

More good news is that there is a proposed technology emerging that can address this issue head-on. More bad news is that a legal quagmire may slow it down, or abort it altogether. The technology is *cryptography,* the standards are described in RFCs 1825–1829, and the quagmire is the restriction on export of cryptographic technology embodied in the law known as the *International Trafficking in Arms Regulations,* commonly referred to as the ITAR.

Cryptography to the rescue

Many current security techniques fall short, sometimes because of generally weak security, sometimes because they cost so much that their implementation is prohibitive, and sometimes because they require explicit user actions (which are prone to being forgotten). Consider two different security problems, authentication and data privacy.

Authentication is the process of determining that you are in fact communicating with the person that you *think* that you are. Many companies require their employees to wear badges showing the corporate logo, the employee's name, and the employee's picture. This way, another employee of the corporation has a reasonable expectation that the person that they are talking to is actually who she says she is (as long as the badge looks authentic and the picture approximately matches what the person looks like).

The problem on a network is how to issue badges. Traditional timesharing systems have long required the use of login sequences combining a publicly known quantity (the username) with an (allegedly) secret quantity (the password). Unfortunately, most networks provide no mechanism to keep the secret password secret, and the entire system collapses if the password becomes known. As a result, it is trivial to capture someone's password using a "sniffer" type attack, and to use the secret password to login as that user.

Some widely available techniques (such as S/key) use a password that is never repeated. This is called a "one-time password," because each password is used exactly once. As a result, there is no value in capturing one of these passwords. However, we aren't safe yet; even though the attacker can't login using our password, she can wait until we login (using our one-time password), and then steal the TCP session. Because the TCP/IP protocols has no mechanism to ensure authentication, anyone on the net can masquerade as us [2]. This is harder to do, but very damaging, since the attacker is now allowed into your network as an authenticated user.

Data Privacy tries to keep your password from being stolen in the first place. Many programs encrypt sensitive information like passwords, and there are a number of file encryptors available on the net (Phil Zimmerman's *Pretty Good Privacy* (PGP) [10] being both the best and the most popular). All of these use one of the many types of cryptography; after all, cryptography is primarily concerned with scrambling (or de-scrambling) the meaning of messages. Unfortunately, each of these techniques requires explicit user action; my cryptosystem only protects my data if I remember to tell it to encrypt the data.

Modular standards, flexible solutions

Cryptographic techniques exist to solve both of these problems. *One-way hash* algorithms act as a kind of cryptographic checksum: when used with a secret key, it is not feasible for someone to modify a message without detection. *Encryption ciphers* change the original message (the *plaintext*) into unreadable gobbledygook (the *ciphertext*). Certain key exchange schemes provide strong authentication. Combining these cryptographic building blocks can give exceptionally strong results. For example, authentication can be provided by a secure key exchange combined with a keyed one-way hash function performed on each data packet. [11, 12].

Examples of one-way hash algorithms are the *Digital Signature Standard* (DSS), which contains the *Secure Hash Algorithm* (SHA). Another algorithm widely implemented is *Message Digest #5* (MD5), developed by RSA Data Security, Inc., and placed in the public domain. Numerous encryption ciphers exist, including the *Data Encryption Standard* (DES) and the *International Data Encryption Algorithm* (IDEA). [3] discusses both privacy and authentication uses of cryptography.

The security architecture proposed by [4] does exactly this. By combining well understood cryptographic techniques, it provides extremely robust security services at the IP layer. Since this architecture is modular, you can implement only those portions that you think are necessary. Since cryptography is restricted in many countries, you can implement only those portions that do not violate local laws. As we will see later, this flexibility is a great asset when addressing the legal aspects of cryptography.

Quo Vadis?: The Authentication Header

Strong authentication is provided by the *Authentication Header* (AH), which is found in the datagram after the IP header and before the Upper Layer Header (e.g., TCP). The same header format is used for both IP version 4 (Classic IP) and IP version 6 (IPng). Figure 1 shows the format of the Authentication Header and its relation to the IP v4 and IP v6 datagrams. [5] defines the Authentication Header.

Format of the Authetication Header

Next Header	Length	Reserved
Security Parameters Index (SPI)		
Authentication Data (variable number of 32-bit words)		

AH for IPv4

IP v4 Header	Auth Header	Upper Protocol (TCP, UDP)

AH for IPv6

IP v6 Header	Hop-by-Hop/Routing	Auth Header	Others	Upper Protocol (TCP, UDP)

Figure 1: The Authentication Header

continued on next page

Towards Real Internet Security *(continued)*

The AH is considered by IP to be protocol number 51 (decimal). The *Next Header* specifies the upper layer header (e.g., TCP or UDP) for IP version 4, or the next header that would be contained in an IP version 6 datagram. The values for these headers are those specified in the "Assigned Numbers" RFC. The length of the AH (in 32-bit words) must be specified, since different authentication techniques produce data of different size, and because data is sometimes padded for performance reasons. *The Security Parameters Index* (SPI) is a pseudo random value used as a reference to any security related information, such as encryption keys. The SPI is unidirectional (i.e., half-duplex), so a typical exchange between two nodes will use a pair of SPIs, one for communications from A to B, and the other for communications from B to A.

Finally comes the specific data (for example, the hash residue produced by the authentication mechanism, e.g., MD5). This can vary (implementation dependent) depending on the data padding used. If the receiving host performs the same authentication check on the datagram, any changes inserted by an attacker will be detected.

Note that at least one of the two partners in the authenticated session may be routers or firewalls (intermediate nodes). This is likely to be the case when only a portion of an organization's computers can be upgraded to the new standards, and it is desired to have an authenticated exchange with one of the non-compliant hosts. In this case, care must be used to ensure that the datagram is not fragmented, since the router will not reassemble the fragments, and will therefore be unable to verify that the AH information matches the datagram.

Implementation

Figure 2 shows several ways that authentication services might be implemented. An internal host is directly authenticating a remote host, a different remote host is authenticating itself to the corporate firewall (either for access control purposes or because it desires an authenticated session with an internal host that does not support the AH), and a hacker is unsuccessfully attempting to penetrate the corporate defenses. This attack fails because it is computationally infeasible for the attacker to guess the 128-bit MD5 key. Note that the hacker could attempt to bypass the firewall, and communicate directly with the end internal host. This too would fail, either because the cryptographic authentication fails, or because the internal host does not support the AH, and rejects the packets.

This Internet standard authentication mechanism should greatly simplify firewalls, and possibly considerably improve their performance. Since firewalls are primarily concerned with authenticating remote devices, the AH represents a major advance in firewall technology.

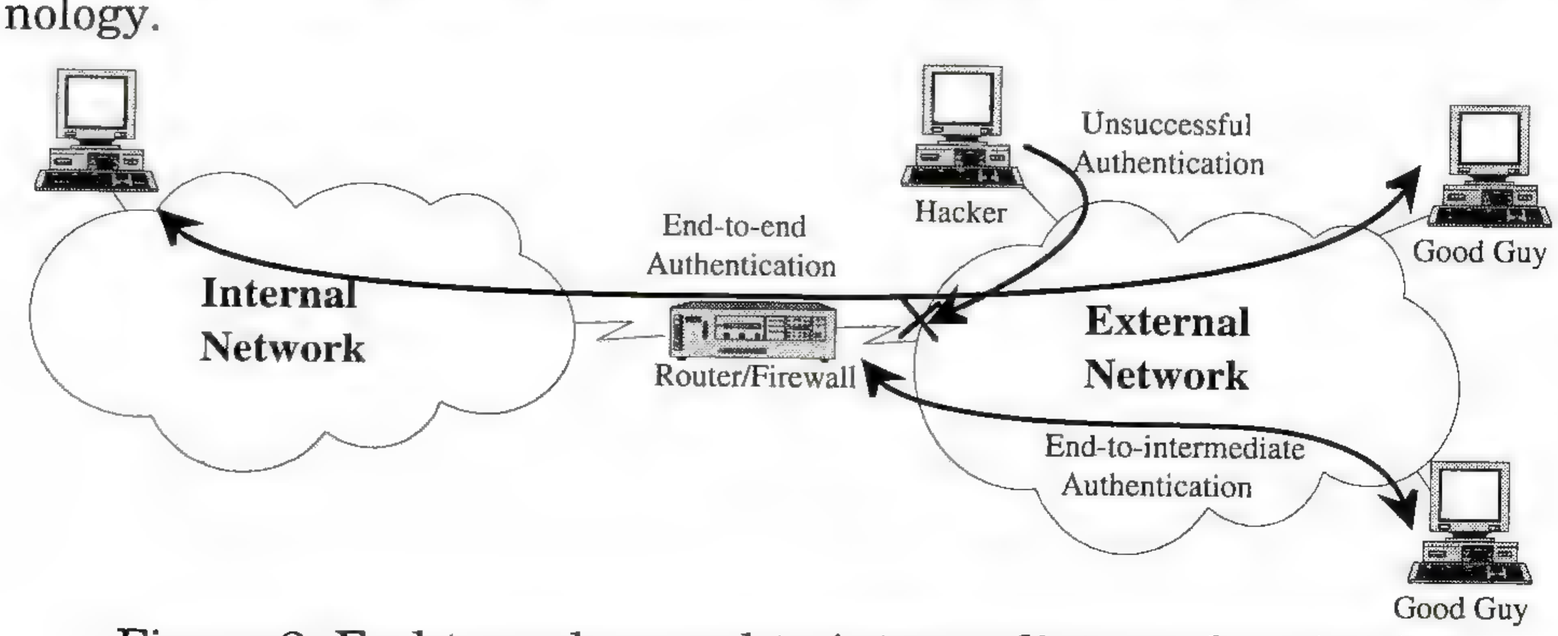


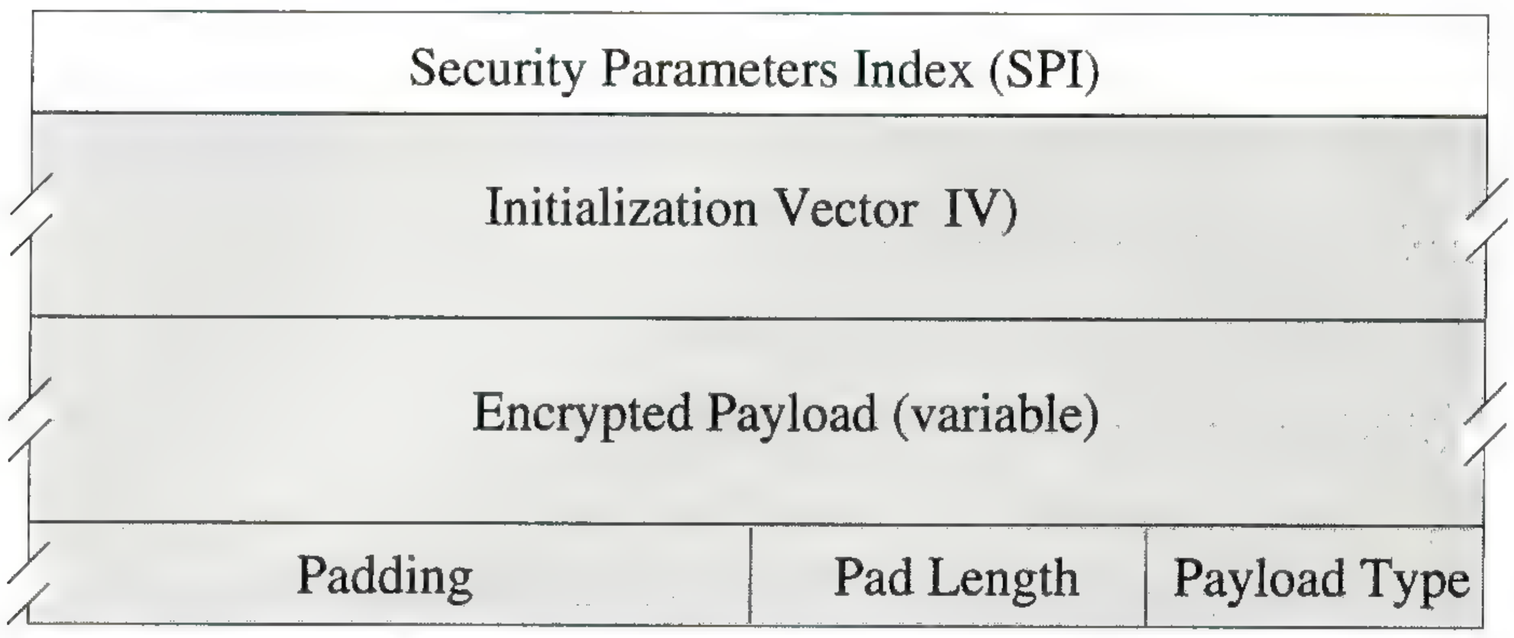
Figure 2: End-to-end vs. end-to-intermediate authentication

[7] specifies how to use keyed MD5 (the MD5 one-way hash function in combination with a secret key) with the AH. This is the only authentication mechanism defined, but it is at least conceivable that several other techniques—such as the DSS—could also be defined to provide authentication services.

What did you just say?: The Encapsulating Security Payload header

Datagrams authenticated by an AH are still plaintext, meaning that the contents is available for inspection by anyone along the data path. For those people who want to exchange trade secrets or other sensitive data, this is insufficient. Therefore, [6] defines a method to encrypt data, rather than just authenticate it. Encryption services are provided via the *Encapsulating Security Payload* (ESP).

ESP services can be provided two ways. The simplest is *transport mode*—the simple encryption of the transport header plus the application data. More complicated (and more interesting) is *tunnel mode*—the capability of encrypting and encapsulating the entire IP datagram. Figure 3 shows the format of the EPS and its relation to the IP headers. Note that only IP version 4 is shown, for simplicity. However, the ESP will also work with IP version 6. The ESP is considered by IP to be protocol number 50 (decimal).



Format of ESP Header for DES–CBC Encryption

IP v4 Header	ESP Header	ULP (TCP, UDP)header plus data

ESP Header for Transport Mode

IP v4 Header	ESP Header	IP v4 header plus ULP (TCP, UDP)header plus data

ESP Header for Tunnel Mode

Figure 3: The Encapsulating Security Payload header

The ESP header is dependent on the particular cipher used to encrypt the data. A Security Parameters Index (SPI, identical in form and function to that in the AH) is followed by an Initialization Vector, the encrypted data, and padding. The encrypted payload is the transport portion of a datagram (e.g., the TCP header and application data) if you are using transport mode, or it is an entire IP datagram (IP header, transport header, application data) if you are using tunnel mode. Note that everything but the SPI is encrypted.

continued on next page

Towards Real Internet Security *(continued)*

Transport mode saves you 20 bytes per datagram, because you are not using a second (encapsulating) IP header. Transport mode ensures privacy for any application that invokes it, saving us the trouble of implementing redundant security services for each application protocol. Implementing the service once, at the transport layer, allows us to forgo implementing Privacy Enhanced Mail, Privacy Enhanced SNMP, and all other Privacy Enhanced *Foo* protocols. We should expect a much wider implementation of security, due to the lessened programming effort that each vendor needs to invest.

Virtual Private Networks

Tunnel mode has two distinct advantages over transport mode. First, it can make traffic analysis next to impossible, if implemented in a corporate firewall or gateway (the firewall would use the "external" address in the outer IP header, and all of the internal addresses will be encrypted—and therefore unreadable). Second, the tunneling capability forms the basis of "Virtual Private Networks"—VPNs. A VPN is the ultimate answer to people who want easy remote connectivity, but don't want the security hassles associated with the Internet: a VPN uses the Internet for transport of data, but doesn't necessarily interact with the Internet as a community. An example of a VPN would be replacing a corporate Frame Relay network with encrypted links running over the Internet, in tunnel mode.

One of the most interesting possibilities provided by tunnel mode is creating non-IP VPNs over the Internet. The standard tunneling technique would let us link isolated pockets of IPX, AppleTalk, or other non-IP protocols, using the Internet only as a transport network.

Figure 4 shows how someone could use the ESP to provide data privacy services, and Figure 5 shows how someone could use the ESP to build a VPN using the Internet as a transport network.

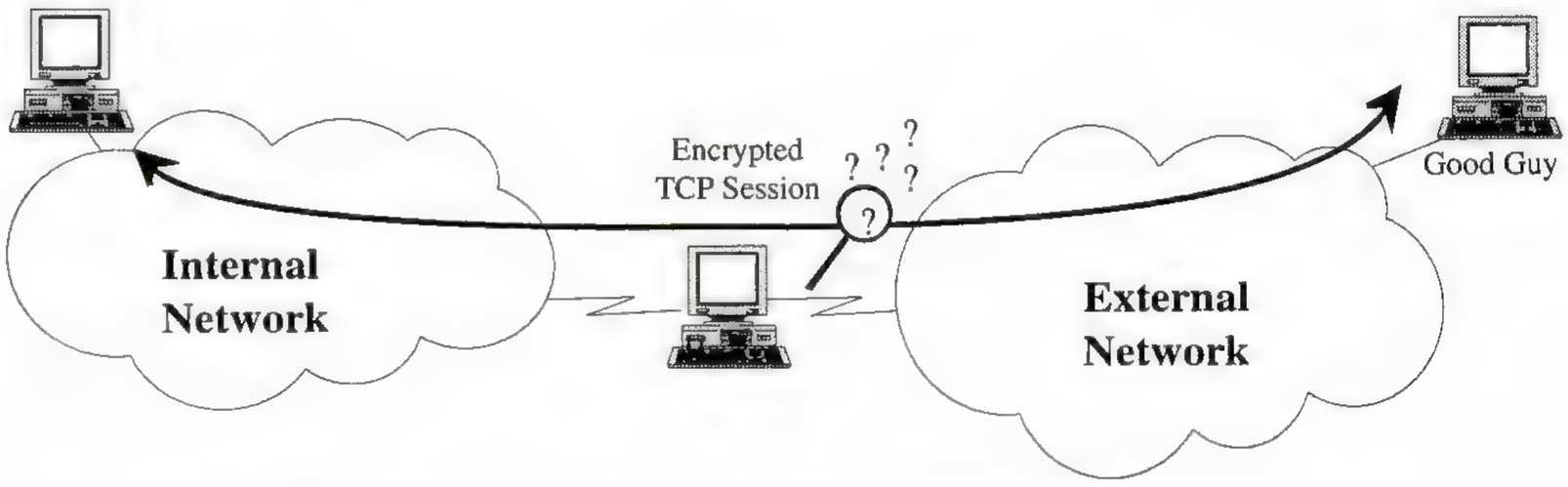


Figure 4: Security services of Transport Mode.

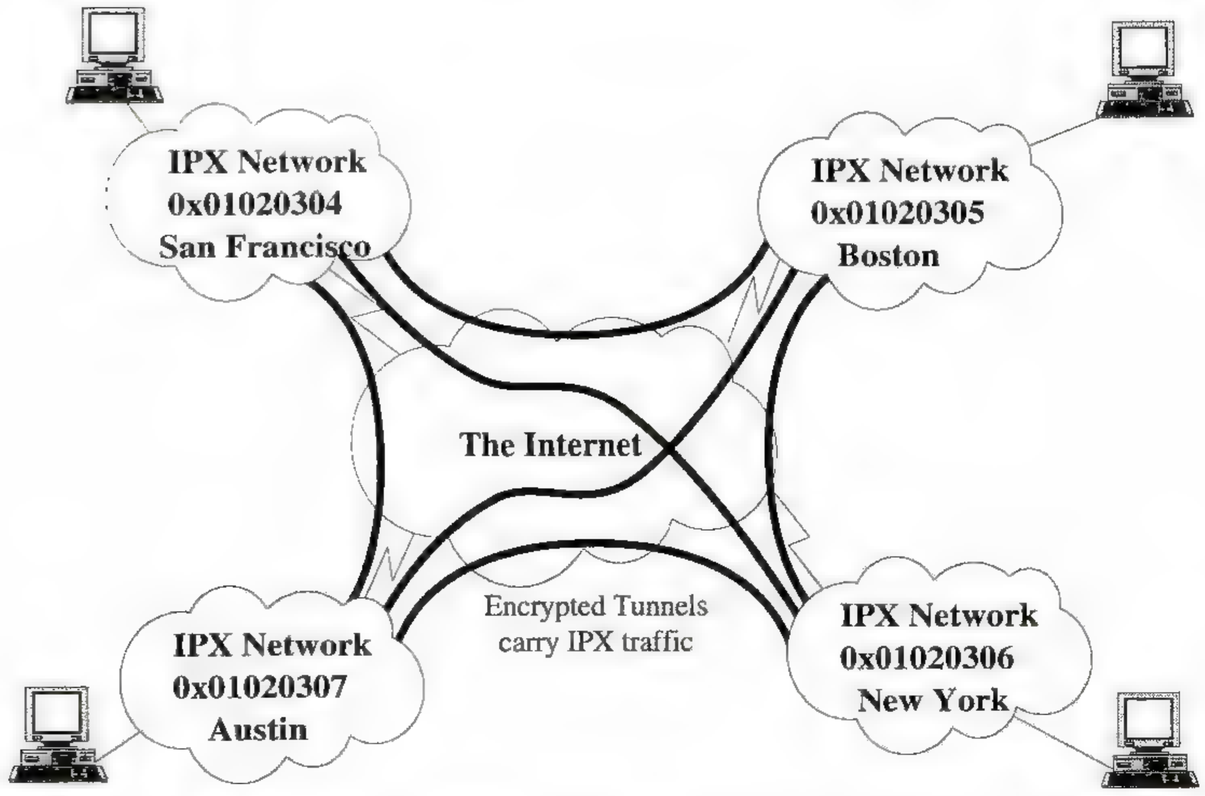


Figure 5: A Virtual Private Network via Tunnel Mode.

[8] describes the use of DES in *Cipher Block Chaining* (CBC) mode with the ESP. As with the one-way hash functions for the Authentication Header, this is the only transform allowed. However, it is possible that other encryption ciphers might be proposed in future RFCs, although this bends the architecture somewhat. Likely ciphers might include Triple DES, IDEA, and Skipjack (as implemented in the Clipper chip, via the FORTEZZA card).

Key Management: Not quite ready for Prime Time

One of the most important areas of cryptography is *key management,* which tries to distribute secret variables used by the encryption or one-way hash algorithms. It is not an exaggeration to say that this is the prime area of possible weakness for any cryptosystem—any attacker able to subvert the key distribution process is very likely to be able to read or forge data sent through the "secure" channel. As a result, great care is being spent on this part of the security architecture.

At this time, the *Internet Key Management Protocol* (IKMP) is still incomplete. There are several efforts (Photuris, SKIP, and ISAKMP) that have been proposed. It is likely that ideas from several will be incorporated in the final standard. As I write this (November 1995), trials of AH and ESP are nearing; however, these will use manual keying. By the spring of 1996, the key management standards are likely to be approaching completion.

The Quagmire: Can IPSec survive the ITAR?

This seems all very positive. Unfortunately, there are powerful interests—mostly law enforcement and intelligence agencies—that are determined that high strength cryptosystems will not be widely implemented (at least not without the government's ability to break them at will). The export of cryptography from most industrialized countries has traditionally been controlled under the *Cocom treaty,* an agreement that defined cryptographic techniques as weapons of war. The laws implementing the Cocom treaty in the United States are known as the ITAR. The ITAR made it a serious offense to ship strong cryptographic products outside the United States or Canada. This has dampened the willingness of vendors to offer encryption products.

Unfortunately (for government control of encryption), the genie is struggling to escape the bottle. Phil Zimmerman's PGP brought very good file encryption to the masses. The Swiss banks developed IDEA as a replacement for the aging DES. Many governments are worried that they will soon be unable to monitor the communications of people inside (or outside) their borders. As a result, many have begun to tighten restrictions on cryptographic products. In the United States, the Clinton administration tried a direct approach to controlling public encryption by mandating the use of the Clipper chip, which relies on a classified encryption cipher (Skipjack). When this turned into a firestorm of controversy, they changed to an indirect attack, by requiring the use of *key escrow.*

Key escrow is a technique that would allow you to use any cipher you like as long as the government can get a copy of your encryption key. While this would allow the reading of confidential communications by law enforcement agencies (presumably with a court order) and intelligence agencies (presumably without a court order), civil libertarians are concerned over the possibility of governments misusing this ability to infringe the privacy of their citizens. This topic is hotly debated, with those taking the government's side talking about protecting citizens from child pornography, and those on the other side talking about personal freedoms being trampled by rogue government agencies (Marcus Ranum goes so far as to say "government mandated key escrow is un-American" [9]).

continued on next page

Towards Real Internet Security *(continued)*

Whether key escrow will delay an Internet standard key management protocol remains to be seen. The issue rouses such passion that even if the Internet standard required key escrow, a group of researchers might propose a different protocol, which did not include it. The next few months promise to be very interesting, as key management moves towards final standard status. Until the dust settles, it is conceivable that vendors will offer support for the Authentication Header, but not for the Encapsulating Security Payload; this way, they could avoid the key escrow controversy and ship product containing strong authentication—but no encryption.

References

[1] Bellovin, S.; "Problems in the TCP/IP Protocol Suite," *Computer Communications Review,* Volume 19, No. 2, April 1989.

[2] CERT Advisory 95-01, "Hijacked Terminal Sessions and IP Address Spoofing." The Computer Emergency Response Team. Available via anonymous FTP from host `cert.org` in: `pub/cert_advisories/CA-95:01.*`

[3] Schneier, B., *Applied Cryptography,* John Wiley & Sons, New York, NY, 1994, ISBN 0-471-59756-2.

[4] Atkinson, R., "Security Architecture for the Internet Protocol," RFC 1825, August 1995.

[5] Atkinson, R., "IP Authentication Header," RFC 1826, August 1995.

[6] Atkinson, R., "IP Encapsulating Security Payload (ESP)," RFC 1827, August 1995.

[7] Metzger, P., and Simpson, W., "IP Authentication using Keyed MD5," RFC 1828, August 1995.

[8] Karn, P. et al., "The ESP DES–CBC Transform," RFC 1829, August 1995.

[9] Private communication, November 24, 1995.

[10] Stallings, W., "Pretty Good Privacy," *ConneXions,* Volume 8, No. 12, December 1994.

[11] Stallings, W., "Cryptographic Algorithms, Part I: Conventional Cryptography," *ConneXions,* Volume 8, No. 9, September 1994.

[12] Stallings, W., "Cryptographic Algorithms, Part II: Public-Key Encryption and Secure Hash Functions," *ConneXions,* Volume 8, No. 10, October 1994.

[13] Dern, D., "Interview with Steve Kent on Internet Security," *ConneXions,* Volume 4, No. 2, February 1990.

[14] Schiller, J., "Issues in Internet Security," *ConneXions,* Volume 7, No. 9, September 1993.

[15] *ConneXions,* Volume 4, No. 8, August 1990, "Special Issue on Network Management and Network Security."

[16] Kaliski, B., "An Overview of Public-Key Cryptography Standards," *ConneXions,* Volume 6, No. 5, May 1992.

TED DOTY stumbled into network security over ten years ago, and has worked as a developer, designing and implementing network protocols; as a systems engineer, integrating security products into Local and Wide Area networks; and as a programmer, implementing large scale systems of packet filters. He is now program manager for security products at Network Systems, working on that company's next generation of security products. His e-mail address is: `ted.doty@network.com`

Call for Papers

The *6th International Workshop on Network and Operating Systems Support for Digital Audio and Video* (NOSSDAV 96) will be held April 23–26, 1996 at the Shonan Village International Conference Center, Zushi, Japan.

Topics

NOSSDAV 96 is the international workshop among active researchers and practitioners who are building innovative multimedia systems, networks and applications. While we will focus on the state of the art technology in networking and operating system support for multimedia systems, we will also seek for practitioners' papers from a variety of area, including media toolkit, mobile communications, VR, real-time systems, software agents, digital library, and distributed computing systems. It is also intended to provide extensive discussion periods during the workshop to discuss the important issues which may require future research. Topics for the workshop include:

- High-speed/ATM networks
- Multimedia-oriented desk, local and wide area networks
- Cell-based system architectures
- Mobile systems for multimedia
- Communication protocols for multimedia
- Multicast protocols and media scaling
- Resource management and reservation in the OS and network
- End-to-end admission control
- Quality of service and synchronization frameworks
- Multimedia storage, server, and I/O architectures
- Distributed multimedia systems
- APIs and CM programming abstractions for multimedia
- TV set-top device communication
- VOD system architecture
- Software agents for multimedia systems
- VR systems

Submissions

Two types of submissions are solicited: *position papers* and *research papers*. For the purpose of paper review, position papers are restricted to three single-spaced ASCII pages. Research papers are restricted to an extended abstract no longer than five formatted *PostScript* pages. Papers should be e-mailed to: `nossdav96@sfc.keio.ac.jp`. Only if electronic submission is impossible, papers may be sent to the following address:

Prof. Hideyuki Tokuda
Keio University
5322 Endoh, Fujisawa
JAPAN 252
Phone: +81-466-47-5000 (wait for 2 seconds, then 3129)
Fax: +81-466-47-0835
E-mail: `hxt@sfc.keio.ac.jp`

The proceedings of the workshop will be published by Springer-Verlag and the best papers will be forwarded to selected journals for publication.

Important dates

Submission deadline:	January 5, 1996
Acceptance notification:	February 12, 1996
Final papers due:	March 18, 1996

More information

`http://www.sfc.wide.ad.jp/nossdav96/`

Call for Papers

The *Third International Workshop on Services in Distributed and Networked Environments* (SDNE '96) will be held in Macau June 3–4, 1996. SDNE '96 is organized in conjunction with the 16th International Conference on Distributed Computing Systems (ICDCS-16) in Hong Kong May 27–30, 1996. The event is sponsored by the IEEE Computer Society Technical Committee on Distributed Processing.

About SDNE '96

SDNE workshops augment the ICDCS program by focusing on global, network-based services and addressing the emerging area of service engineering, building on international standards such as ANSA, ODP, DCE, CORBA, and TINA. These layers are the middleware that glue applications to the distributed environment, insulating them from location dependencies where desirable, alerting them to location information where necessary. Usability and usefulness of network services depends upon the kind and quality of software services provided to the users, availability of information on existing resources, ease of developing new applications, reliability, and security.

SDNE '96 builds on the success of the First and Second International Workshops on Services in Distributed and Networked Environments (SDNE '94, Prague, Czech Republic; SDNE '95, Whistler, British Columbia, Canada). The international flavor of the workshop reflects the scope and diversity of worldwide internetworking. Past SDNE workshops have had representation from North and South America, Europe, Australia, Asia, and the Middle East.

About Macau

Macau is a Portuguese territory located in the south of China on the West bank of the Pearl River estuary, 64 kilometers from Hong Kong. Macau features a wealth of fascinating and historical monuments, museums, fortresses, and exciting tourist activities, packed into 16 square kilometers.

Topics

The workshop seeks original papers on topics related to providing services to applications, ranging from web search engines to RPC to algorithms for multicast, and beyond. Past SDNE sessions have been devoted to mobile services, collaboration, Internet information services, programming models for service engineering, and many others. The workshop is targeted to be a forum for free flow of ideas. Reports on experimental work are particularly welcome. Presentations of new ideas and work in progress are also invited. The SDNE workshop seeks submissions in the following areas:

- Protocols and abstractions for service engineering, management, brokerage
- Case studies of service creation and service deployment
- Internet services (archie, gopher, netfind, Prospero, WAIS, WWW, etc.)
- Mobile computing and services for mobile users
- Client–server programming, RPCs, and service strategies
- Security, accounting, and management services
- Using objects for distributed services
- Persistence and concurrency in distributed services
- DCE, CORBA, and ANSAware-based services
- Interworking of heterogeneous services

- Structure, usability, and performance of distributed services
- Quality of service aspects of networked environments
- Information retrieval/location services for large scale networks
- Multicast and scalable services
- Naming and directory services
- Electronic commerce protocols and services
- Broadband services to homes and telecommuters

Submission guidelines

You are invited to submit a full paper in English for presentation at SDNE '96. There will be an eight page limit on published papers; submitted papers should be about that length. All submissions will be reviewed by the Program Committee. Papers accepted for presentation at SDNE '96 will be included in the proceedings distributed at the workshop and made available from IEEE.

Electronic submission in *PostScript* is strongly encouraged. Please include an abstract and a cover page with address, telephone, fax and e-mail of the primary contact person. Submissions should be sent to:

Peter Honeyman
Center for Information Technology Integration
University of Michigan
519 W. William St.
Ann Arbor, MI 48104–4943
USA
E-Mail: `sdne96@citi.umich.edu`
Tel : +1 313 763 4413
Fax : +1 313 763 4434

Further information may also be obtained from the above address.
This call is available at: `http://www.citi.umich.edu/sdne.html`

Important dates

Submissions due:	February 1, 1996
Notification by:	March 1, 1996
Full papers due:	April 8, 1996

Organization

Nigel Davies, Lancaster University, UK, *General Chair*
Peter Honeyman, University of Michigan, USA, *Program Chair*
Robert P. Biuk-Aghai, Univ. of Macau, *Local Arrangements Chair*

Program Committee

Jean Bacon, Cambridge University, UK
Ashley Beitz, DSTC, Australia
Mark E. Crovella, Boston University, USA
David De Roure, University of Southampton, UK
Elmootazbellah Elnozahy, Carnegie Mellon University, USA
Markus Endler, University of Sao Paulo, Brazil
Jan Janecek, Czech Technical University, Czech Republic
Thomas Koch, University of Hagen, Germany
Rodger Lea, Sony Corporation, Japan
Gerald Neufeld, University of British Columbia, Canada
Stephen Pink, SICS, Sweden
Herman Rao, AT&T Bell Labs, USA
John Rosenberg, University of Sydney, Australia
Rich Salz, OSF, USA
Alexander Schill, Technical University of Dresden, Germany
Ellen Siegel, Sun Microsystems, USA
Morris Sloman, Imperial College, UK
Paulo Verissimo, INESC, Portugal

CONNEXIONS
303 Vintage Park Drive
Suite 201
Foster City, CA 94404-1138
Phone: 415-578-6900
FAX: 415-525-0194

ADDRESS CORRECTION
REQUESTED

FIRST CLASS MAIL
U.S. POSTAGE
PAID
SAN JOSE, CA
PERMIT NO. 1

CONNEXIONS

EDITOR and PUBLISHER Ole J. Jacobsen